DWKS : A Local Descriptor of Deformations Between Meshes and Point Clouds

Robin Magnet
LIX, École Polytechnique
rmagnet@lix.polytechnique.fr

Maks Ovsjanikov
LIX, École Polytechnique
maks@lix.polytechnique.fr

Abstract

We propose a novel pointwise descriptor, called DWKS, aimed at finding correspondences across two deformable shape collections. Unlike the majority of existing descriptors, rather than capturing local geometry, DWKS captures the deformation *around a point within a collection in a multi-scale and informative manner. This, in turn, allows to compute inter-collection correspondences without using landmarks. To this end, we build upon the successful spectral WKS descriptors, but rather than using the Laplace-Beltrami operator, show that a similar construction can be performed on shape difference operators, that capture differences or distortion within a collection. By leveraging the collection information our descriptor facilitates difficult non-rigid shape matching tasks, even in the presence of strong partiality and significant deformations. We demonstrate the utility of our approach across a range of challenging matching problems on both meshes and point clouds. The code for this paper can be found at* https://github.com/RobinMagnet/DWKS

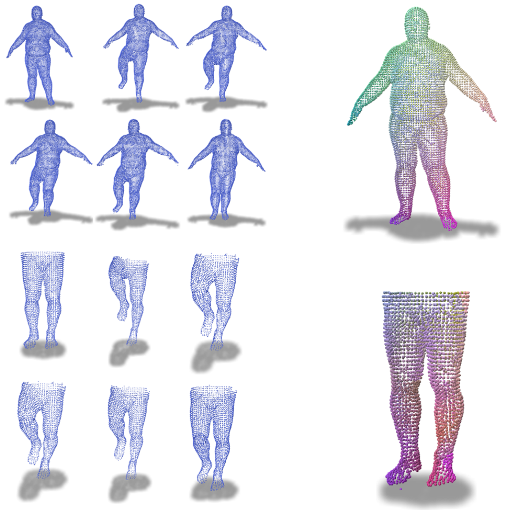

Figure 1. Our method uses two collections of noisy point-clouds with roughly similar deformations (left) and outputs a point-wise inter-collection map (right).

1. Introduction

Shape matching is an ubiquitous problem in 3D computer vision, with various applications like texture and deformation transfer. Numerous methods have therefore been developed during the last decade to compute correspondences between surfaces, relying on simple rigid deformations to more recent learning-based models [44, 36].

While these methods have shown impressive results on several datasets, there remain some very challenging scenarios especially when dealing with symmetries and non-isometric shapes. The existence of intrinsic symmetries in non-rigid shapes (*e.g.* left-right symmetry in a human shape) can be handled in multiple ways, using an orientation preserving constraint on shape descriptors [31], exploring the space of maps [30], or adding priors through learning-based methods [23, 34, 9]. These methods respectively require the existence of precise shape descriptors, an automatic selection of symmetric and non-symmetric map, and large datasets for neural network training. The case of non-isometric shapes is typically addressed either by requiring user-specified landmarks [1, 11, 38] or again, through extensive neural network training [14], among many other approaches. In the specific case of *partial* non-rigid shape matching, several methods have been developed [33, 24], relying on theoretical properties of the changes to the Laplace-Beltrami operator under partiality.

Remarkably while 3D shapes often come in the context of a collection, very few methods [39, 5] have tried to leverage the commonality of *the deformations* that exist within the collections to facilitate matching across them, *e.g.* to disambiguate such symmetries or address partiality. Notably while matching two human shapes in resting pose can suffer from their intrinsic symmetry ambiguity, using information from the deformed version of these shapes with *e.g.*

their left knee up could help disambiguate their symmetries. Figure 1 exhibits how local deformations of the knees when jumping on one leg allow to compute correspondences between two human shapes, even when the upper-half of the body is missing.

In this work we propose to develop a local descriptor of differences between surfaces and point clouds. Our method is inspired by successful spectral point-based descriptors extracted from the Laplace-Beltrami operator [43, 2]. Our main insight is that a similar construction can be performed on other functional operators, leading to informative descriptors that capture different properties of shapes and collections. In our work, we use shape difference operators [35] that have been used for both analysing deformations within a collection [16] and even shape synthesis [19]. In the context of cross-collection mapping, shape difference operators have been used in [39, 5] as global objectives within the functional map framework, which can limit their utility to *complete* shape matching. Instead, we demonstrate that pointwise spectral descriptors can successfully be extracted from shape difference operators. Our descriptor, termed DWKS, thus combines the power and flexibility of local descriptors with the information of shape distortion present in shape difference operators, which, as we demonstrate below, makes it applicable in partial cross-collection matching scenarios. We also exploit recent advances in constructing robust operators [41] to enable accurate and efficient matching across shapes represented as both meshes and point clouds.

Our main contributions can be summarized as follows: 1. we introduce a novel pointwise descriptor that reflects deformation around a point within a collection, 2. we demonstrate how spectral methods, and specifically the WKS desciptors can be extended beyond the Laplacian to shape difference operators, and 3. we demonstrate how difficult matching scenarios with partiality and symmetry ambiguity, on both point clouds and meshes can strongly benefit from our descriptor without landmarks or neural network training.

2. Related Work

The shape matching literature is very vast and we will only highlight existing methods that are most relevant to our setting. We refer the reader to a recent survey [37] on the subject for more information. We base our method on the functional map framework defined in [28] which seeks to match functional spaces on the shapes instead of the shapes themselves, and has led to impressive results in the last decade. Several follow-up works [27, 45, 31, 18, 25, 30] have brought substantial improvements on the original pipeline, and all heavily rely on the existence of consistent descriptor functions of shapes which are functions supposed to be preserved by the mapping, based either on local descriptors [43, 2, 8] or landmarks. Generating informative and robust descriptors in a fully automatic way remains a very challenging problem, and often requires near-isometric shapes without symmetries. To alleviate this issue, recent works have sought to learn descriptors using neural networks, either from usual descriptors [23, 34] or directly from raw data [9, 40]. This allows to incorporate prior information into descriptors, and possibly disambiguate symmetries like left and right for a human shape.

A more demanding setting lies in partial shape matching, which is a simple case of non-isometric shape matching. A remarkable adaptation of the original framework was introduced in [33, 24], based on the theoretical insights of the relation between the Laplace-Beltrami operators of a shape and a compact subset of it.

Methods computing correspondences using functional maps can typically produce somewhat noisy correspondences which then serve as initialization for refinement algorithms. The original refinement technique uses a variant the ICP algorithm [28] which supposes shapes to be isometric, but new more general approaches have then been developed [10, 31, 11, 25, 20]. The ZoomOut algorithm [25] is of particular interest as it starts from very rough correspondences to obtain high quality maps through spectral upsampling. While it also relies on a strong near-isometry assumption, the theoretical background developed in [33] can be used to adapt the algorithm to partial shapes in practice.

More related to our contribution are multiple works on shape collections. Several methods have been developed to refine intra-collection correspondences using cycle consistency constraints, *e.g.* [26, 15, 42, 20, 12] among many others. These methods also leverage information within shape collections, but are typically not aimed at computing cross-collection maps and often still rely on pairwise map estimation as a building block. Extracting information about the variability of shapes within a collection was brought about by the introduction of shape difference operators [35], which summarize intrinsic distortion between a pair or a collection of shapes as two functional operators using simply rough correspondences between two shapes represented as functional maps. These two “difference” operators, together with a source shape, have been shown to be sufficient to reconstruct the deformed version up to isometric deformations [6], and up to rigid motion using additional extrinsic shape differences [6, 19]. Shape difference operators provide a powerful tool for summarizing the variability within shape collections, which has motivated their use in computing cross-collection shape correspondences. Our work is directly inspired by the excellent results shown in [39, 5] where corresponding shape difference operators are matched together to compute cross-collection functional maps. The solving procedure, however, relies on SVD which suffers both from sign ambiguity and possible instability. Furthermore the method supposes the global deformations to be matched to correspond, which breaks down in the case of partiality.

In this work we focus on computing local or pointwise descriptors from shape differences, which can be used both within the functional maps pipeline and beyond [2, 28]. We show that while shape difference operators [35] capture the global difference between shapes, their properties allow pointwise information to be extracted in the form of vertex-wise descriptors. This information can be used either in conjunction with the pipeline of [39, 5] in the case of complete shapes, or even directly, in the case of partial shapes

3. Background

3.1. Functional Maps

Our work falls within the functional map framework originally introduced in [28] and that we review below briefly for completeness. Given two surfaces $\mathcal{M}$ and $\mathcal{N}$, a point-wise correspondence $T : \mathcal{N} \to \mathcal{M}$ can be equivalently represented as a linear (functional) map $F : L^2(\mathcal{M}) \to L^2(\mathcal{N})$ between the space of squared integrable functions on each shape.

Using an appropriate basis for the two functional spaces, the functional map F can be represented as a possibly infinite matrix. Specifically, eigenfunctions of the Laplace-Beltrami operator of each shape have had a lot of success in spectral shape analysis [22, 32, 28] and can be interpreted as Fourier basis for functions on surfaces, and enable basis truncation due to their multi-scale nature.

3.2. Shape Matching

The standard functional correspondence pipeline [29] between shapes $\mathcal{M}$ and $\mathcal{N}$ looks for a functional map $\mathbf{C} \in \mathbb{R}^{k_\mathcal{N} \times k_\mathcal{M}}$ from $L^2(\mathcal{M})$ to $L^2(\mathcal{N})$, where $k_\mathcal{M}$ and $k_\mathcal{N}$ represent the size of the corresponding (truncated) basis.

Given a set of descriptor functions on each shape $\{(f_i, g_i)\}_{i=1}^p$ with $f_i \in L^2(\mathcal{M})$ and $g_i \in L^2(\mathcal{N})$, expected to be preserved under the functional map, we encode them in their respective basis as two matrices $\mathbf{A} \in \mathbb{R}^{k_\mathcal{M} \times p}$ and $\mathbf{B} \in \mathbb{R}^{k_\mathcal{N} \times p}$. Standard choices are HKS [43] or WKS [2] descriptors.

Denoting $\Delta^\mathcal{M}$ (resp. $\Delta^\mathcal{N}$) the Laplace-Beltrami operator on shape $\mathcal{M}$ (resp. $\mathcal{N}$), expressed in their respective basis as diagonal matrices, the functional correspondence problem is written as:

$$\underset{\mathbf{C} \in \mathbb{R}^{k_\mathcal{N} \times k_\mathcal{M}}}{\text{argmin}} \quad \|\mathbf{CA} - \mathbf{B}\|_F^2 + \mu_l \|\mathbf{C}\Delta^\mathcal{M} - \Delta^\mathcal{N}\mathbf{C}\|_F^2 \tag{1}$$

with $\|\cdot\|_F$ the Frobenius norm. Here the first term ensures descriptor preservation, while the second one favors isometric maps, and $\mu_l \in \mathbb{R}$ is a manually set scaling factor.

Among many extensions to this basic pipeline, *e.g.* [21, 31, 45, 33, 13] a notable one introduced in [27] and that we use below, proposed a term promoting the functional maps to arise from pointwise correspondences. For this, a functional operator is associated to each input descriptor $\mathbf{\Gamma}_{f_i}$, $\mathbf{\Gamma}_{g_i}$, that acts on other functions through multiplication. These operators are then introduced into the optimization objective (1), by promoting commutativity with them, namely $\mu_{dc} \sum_i \|\mathbf{C}\mathbf{\Gamma}_{f_i} - \mathbf{\Gamma}_{g_i}\mathbf{C}\|_F^2$. While this pipeline can produce accurate correspondences given appropriate descriptors it suffers from multiple issues. Namely, it does not allow to disambiguate symmetries, requires specific adaptation for partial matching and more broadly does not take into account information about *collections* that shapes often naturally are part of.

3.3. Shape Difference Operators

Our work also heavily relies on shape difference operators introduced in [35], that intuitively capture differences or distortion across a pair or within a collection of shapes. Specifically given shapes $\mathcal{M}_1$ and $\mathcal{M}_2$ with known correspondences encoded as a functional map F between them, and inner products $\langle \cdot, \cdot \rangle_{\mathcal{M}_1}$ and $\langle \cdot, \cdot \rangle_{\mathcal{M}_2}$ on each shape, the associated *shape difference operator* is defined as the unique linear operator D acting on $L^2(\mathcal{M}_1)$ so that

$$\langle f, Dg \rangle_{\mathcal{M}_1} = \langle F(f), F(g) \rangle_{\mathcal{M}_2} \quad \forall f, g \in L^2(\mathcal{M}_1) \tag{2}$$

This operator can be seen as compensating the distortion induced by F with respect the given inner products. Note that shape difference operators sharing both a common source shape and inner product can all be compared as they all act on the same functional space.

The original work [35] introduced two shape difference operators, which capture the complete intrinsic distortion across shapes. The first one $V_{\mathcal{M}_1,\mathcal{M}_2}$, is associated to the standard L^2 inner product on both shapes $\langle f, g \rangle_{L^2(\mathcal{S})} = \int_\mathcal{S} f(x) g(x) d\mu^\mathcal{S}$ on a shape $\mathcal{S}$. The second one, denoted $R_{\mathcal{M}_1,\mathcal{M}_2}$ is associated to the H_0^1 inner product $\langle f, g \rangle_{H_0^1(\mathcal{S})} = \langle \nabla f, \nabla g \rangle_{L^2(\mathcal{S})}$.

The two operators $V_{\mathcal{M}_1,\mathcal{M}_2}$ and $R_{\mathcal{M}_1,\mathcal{M}_2}$ are called respectively area-based and conformal shape differences since they equal identity if the underlying maps are respectively area-preserving and conformal [35].

Using the spectral basis of size k_1 and k_2 to encode the F into $\mathbf{C} \in \mathbb{R}^{k_2 \times k_1}$, the shape difference operators can be computed directly as $k_1 \times k_1$ matrices

$$\mathbf{V}_{\mathcal{M},\mathcal{N}} = \mathbf{C}^\top \mathbf{C} \tag{3}$$

$$\mathbf{R}_{\mathcal{M},\mathcal{N}} = \left(\Delta^\mathcal{M}\right)^\dagger \mathbf{C}^\top \Delta^\mathcal{N} \mathbf{C} \tag{4}$$

with † denoting the Moore-Penrose pseudo inverse.

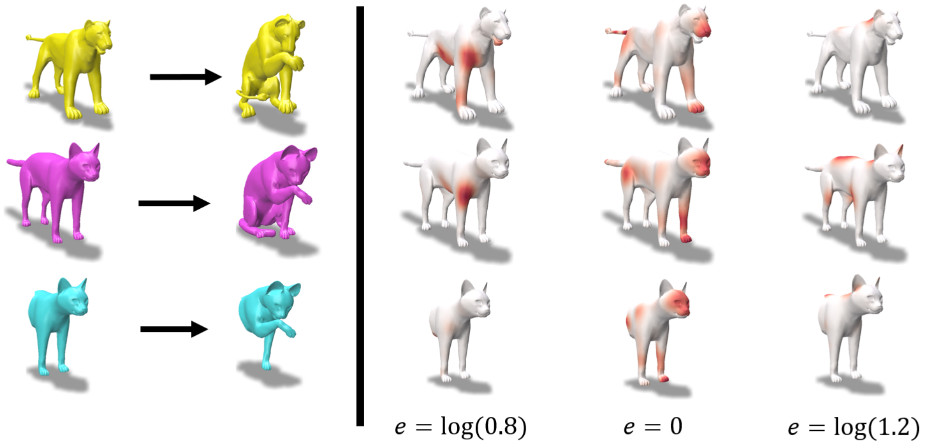

Figure 2. Examples of DWKS descriptors for meshes. The left part displays the source meshes and their deformed version. The right part displays for each mesh the DWKS descriptors at 3 fixed energy levels shown at the bottom (seen as a function of the mesh). Notice that descriptor remain somewhat consistent even in the case of partiality.

3.4. Matching with Shape Difference Operators

Although originally shape difference operators were introduced for shape analysis, they also have been used for solving cross-collection shape correspondence problems [39, 5]. Specifically, given two shape collections $\{\mathcal{M}_i\}_{i=0}^n$ and $\{\mathcal{N}_i\}_{i=0}^n$, where deformation between $\mathcal{M}_0$ and $\mathcal{M}_i$ are similar to deformation between $\mathcal{N}_0$ and $\mathcal{N}_i$ for any $i \in \{1, \dots, n\}$, the goal is to compute a cross-collection map between $\mathcal{M}_0$ and $\mathcal{N}_0$ which we denote as $\mathcal{M}$ and $\mathcal{N}$.

With similar differences between these pairs of shapes, we expect their associated shape difference operators $\mathbf{D}_i^{\mathcal{M}}$ and $\mathbf{D}_i^{\mathcal{N}}$ to act similarly, where $\mathbf{D}$ denotes any of the area or conformal shae difference operator and the index indicates the operator is associated to the deformation between shape 0 and shape i. In the functional map framework, this is equivalent to expecting the sought functional map to commute with these operators, that is $\mathbf{C}\mathbf{D}_i^{\mathcal{M}} \simeq \mathbf{D}_i^{\mathcal{N}}\mathbf{C}$. This leads to the optimization problem solved in the recent approach of [5] (which extends the method in [39]):

$$\underset{\substack{\mathbf{C}\in\mathbb{R}^{k_{\mathcal{N}}\times k_{\mathcal{M}}}\\ \|\mathbf{C}\|_F=1}}{\operatorname{argmin}} \sum_{i=1}^{n} E_i(\mathbf{C}) + \alpha\|\mathbf{C}\Delta^{\mathcal{M}} - \Delta^{\mathcal{N}}\mathbf{C}\|_F^2 \qquad (5)$$

where $\alpha \in \mathbb{R}$ is a scaling factor and

$$E_i(\mathbf{C}) = \|\mathbf{C}\mathbf{R}_i^{\mathcal{M}} - \mathbf{R}_i^{\mathcal{N}}\mathbf{C}\|_F^2 + \|\mathbf{C}\mathbf{V}_i^{\mathcal{M}} - \mathbf{V}_i^{\mathcal{N}}\mathbf{C}\|_F^2 \quad (6)$$

Note that unlike the standard functional matching pipeline (1), the optimization objective (5) does not rely on the existence on coherent descriptors. Moreover, without the constraint $\|C\|_F = 1$ the trivial solution $C = 0$ would give zero error. The authors of [39, 5] solve the problem in Eq.(5) using SVD, which results both in sign ambiguity for the solution and instability in practice. Moreover, the second term of the objective (5) acts as a powerful regularizer in the case of near-isometric shapes but fails in more challenging settings, including partiality.

```
Algorithm 1: Computing DWKS descriptors
  Input: A shape difference operator $\mathbf{D}$ expressed in
    the reduced basis, eigenvectors of the Laplacian on
    the source shape as columns of $\mathbf{\Phi}$, a list of $p$
    energy values $(e_j)_j$, a scale parameter $\sigma$
  Output: One DWKS descriptors at each energy
    value.
  (1) Compute the eigenvectors $\mathbf{U}$ and the eigenvalues
    $(\lambda_i)_i$ of $\mathbf{D}$
  (2) Compute $\mathbf{\Psi} = \mathbf{\Phi}\mathbf{U}$ the eigenvectors of $\mathbf{D}$ in the
    canonical basis.
  (3) Use $\mathbf{\Psi}$, $(\lambda_i)_i$ and $\sigma$ to compute the DWKS
    descriptor for each $e_j$ using equation (7).
```

In this work, we build on this pipeline and use local pointwise descriptors extracted from the shape difference operators. This allows both to use standard optimization techniques, thus avoiding the costly SVD associated with $\|C\|_F = 1$ regularization, and to remove the need for near-isometric regularization. Ultimately, our framework is both more efficient and leads to significant improvements, especially in the case of partial shapes.

4. Our approach – DWKS

4.1. Motivation and Overview

The standard functional correspondence pipeline described in Section 3.2 relies on both commutativity with the Laplacian operators and alignment of local descriptors.

Algorithm 2: Aggregate DWKS descriptors for a collection

```
Input: A list of functional maps $\{\mathbf{C}_i\}_{i=0}^n$ between
  the base shape and shape $i$, eigenvectors $\mathbf{\Phi}$ of the
  Laplacian on the base shape, a list $\{\Delta_i\}$ of
  diagonal matrices of eigenvalues of the Laplacian
  for each shape, energy values $(e_j)_{j=1}^p$, a scale
  parameter $\sigma$
Output: DWKS descriptors for the complete
  collection
for $i \leftarrow 1$ to $n$ do
    Compute $\mathbf{V}_i$ and $\mathbf{R}_i$ using Eq. (3) and Eq. (4)
      with $\mathbf{C}_i$, $\Delta_0$ and $\Delta_i$.
    Compute DWKS descriptors of $\mathbf{V}_i$ and $\mathbf{R}_i$
      using Algorithm 1 with $\sigma$, $\mathbf{\Phi}$ and $(e_j)_{j=1}^p$
end
```

Interestingly, spectral descriptors such as HKS or WKS [43, 4, 2] are extracted from the same Laplacian operators. Nevertheless, their use in the optimization problem of Eq. (1) both helps to prevent trivial solutions and injects local information into the process. Our main goal is to mimic this construction for cross-collection matching, but using shape difference operators. Interestingly, commutativity with shape differences has already been advocated in [39, 5]. We seek to extend this construction by also extracting pointwise descriptors from shape difference operators, similarly to the way WKS is extracted from the Laplacian.

Unfortunately, such an adaptation is not straightforward primarily because unlike the Laplace-Beltrami operators whose spectral properties are well-understood and have intuitive physical interpretations, shape difference operators are much less studied and it is therefore not clear whether pointwise spectral descriptors can be extracted in the same manner. We thus start with the following key observation (with proof given in the supplementary materials):

Theorem 1 *Given a non-degenerate functional map F, both the area-based and conformal shape difference operators are positive (semi)-definite, provided that the area and stiffness matrices of the Laplacian are positive (semi)-definite.*

This theorem, which interestingly was not demonstrated in the original shape difference work [35], provides the first insight into the possibility of applying spectral approaches to shape difference operators, since, similarly to the Laplacian their *eigenvalues are guaranteed to be non-negative*. Moreover, we remark that shape difference operators enjoy both locality and composition or functoriality properties (see, respectively, propositions 4.2.3 and 4.2.4 in [7]). The former remark resonates with the more general property of shape difference operators in [35], which states these two operators act on functions depending on the *local distortion* induced by the underlying correspondence map.

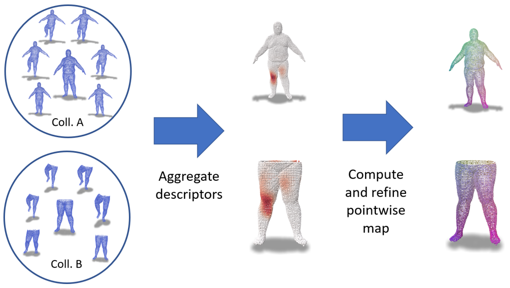

Figure 3. Our pipeline takes two collections as input, with given base shapes. DWKS descriptors for each deformation are aggregated to smooth out the noise, and are then used for point-wise map computation.

While the eigenfunctions of the Laplace-Beltrami operator capture the "smoothest" possible functions on the surface, the eigenfunctions of the shape difference operators, intuitively, capture areas of distortion between the shapes (see [35, 17] for a discussion of this property). Moreover functions that are *preserved* by the shape differences $Df = f$ (and thus correspond to eigenvalue 1) correspond to areas of *no* distortion (see Theorem 4.2.1 in [7]). One can draw a parallel with the constant function, corresponding to the zero eigenvalue of the Laplacian. Moreover, shape differences naturally enjoy the multiplicative algebra [35] (see also Proposition 4.2.4 in [7]), which means, for example that $D_{\mathcal{N},\mathcal{M}} = (D_{\mathcal{M},\mathcal{N}})^{-1}$ and $D_{\mathcal{M},\mathcal{N}} = D_{\mathcal{M},\mathcal{P}} D_{\mathcal{P},\mathcal{N}}$ for any shape difference operator D and shapes $\mathcal{M}$, $\mathcal{N}$ and $\mathcal{P}$ (up to the appropriate change of basis).

The two observations above suggest that the spectrum of shape difference operators is more naturally expressed using the *log-scale*. This way, the undeformed regions correspond to $log(1) = 0$ log-eigenvalues. Moreover, the log-eigenvalues of the operator that captures the inverse deformation are simply negative of that of the direct deformation. Finally, in some cases (*e.g.*, when deformations commute) the composition of difference operators leads to log-eigenvalues being *sums* of individual difference operators. We expand upon these observations and provide a more formal treatment in the supplementary materials.

A final but essential remark is that the shape difference operators we use in practice are all expressed using the truncated basis of Laplace-Beltrami eigenfunctions using Eq. (3) and (4). The eigenfunctions of shape differences can therefore only represent very smooth functions and in particular cannot represent a Dirac delta function on the mesh but rather a heat kernel centered around a point.

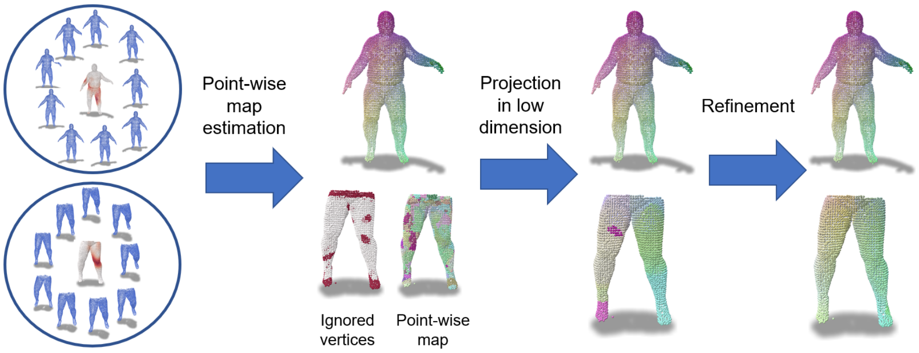

Figure 4. Visualization of the fitting pipeline. Starting from descriptors, a first point-wise map is computed, which is then projected into a low-dimension functional map ignoring some outlier vertices. This functional map is then refined using ZoomOut algorithm [25].

4.2. Definition

Using these remarks and inspired by the definiton of WKS descriptors [2], we define the DWKS descriptor of a given shape difference operator D acting on shape $\mathcal{M}$ as

$$\begin{aligned}\mathrm{DWKS}(D) &: \mathcal{M}\times\mathbb{R}\to\mathbb{R}\\ &(x,e)\mapsto C\sum_{\substack{k=1\\ \lambda_k>0}}^{k_\mathcal{M}} e^{-\frac{(e-\log(\lambda_k))^2}{2\sigma^2}}\psi_k(x)^2\end{aligned} \tag{7}$$

with $(\lambda_k)_k$ and $(\psi_k)_k$ respectively the eigenvalues and eigenvectors of the operator D, σ a manually set parameter, and C ensures that $\int_\mathbb{R}\|\mathrm{DWKS}(D)(\cdot,e)\|_\mathcal{M} de = 1$.

DWKS can be interpreted as a Gaussian blur of the spectrum of the operator, where parameter σ defines the spread of eigenvectors on the $\log$ scale. Remarks from Section 4.1 motivate the choice of a constant σ across all energy levels.

Note that in order to compute a DWKS descriptor we assume to be given either a pair or a collection of shapes with functional maps between them. The functional maps are represented in the truncated Laplacian eigenbasis, which leads to small-size shape difference operator matrices. Note that DWKS also produces a separate pointwise descriptor for each (area-based and conformal) shape difference operator, and can be extended to any shape difference operator by applying the construction described in Algorithm 1.

Examples of DWKS descriptors of the area shape difference operator, seen as functions of the shape at a given energy level are displayed on Figure 2. Each line displays descriptors for similar deformations of a cat and a lion, which do not share either similar geometry or number of vertices. Note that the descriptors seem quite similar up to some noise as with $e = \log 1.2$. The bottom line demonstrates that the descriptors remain stable even in the case of partiality as they capture local information.

4.3. Stability of descriptors

As seen from Figure 2, DWKS descriptors seem to remain stable even in the case of partial shapes, but the reason might be unclear.

When comparing DWKS descriptors between two complete and two partial shapes, two phenomenona occur. On the one hand, eigenfunctions of shape difference operators and therefore the associated DWKS descriptors are more localized in the partial case since the spectral basis can represent more precise elements, as shown in [33]. On the other hand, only a fraction of the less localized eigenfunctions in the complete case are located on the zone represented by the partial shape. Eventually we observe in practice that these two effects get averaged out by the Gaussian blur

In practice DWKS descriptors provide partial information on shape deformation and therefore use additional regularization to obtain more meaningful point-to-point correspondences. In the following we present one possible pipeline, illustrated on Figure 3 to obtain point-wise maps from DWKS descriptors.

4.4. Matching Pipeline

We suppose being given two similar collections of shapes $(\mathcal{M}_i)_{i=0}^n$ and $(\mathcal{N}_i)_{i=0}^n$, aligned in the sense that deformation between $\mathcal{M}_0$ and $\mathcal{M}_i$ is similar to the one between $\mathcal{N}_0$ and $\mathcal{N}_i$ for all i. Note that this information can be automatically retrieved from unaligned collections of different size using the pipeline from [5]. We also assume to have access to approximate intra-collection maps, which can be computed using known near-isometric shape matching technique.

For simplicity we equivalently write $\mathcal{M}$ (resp. $\mathcal{N}$) or $\mathcal{M}_1$ (resp. $\mathcal{N}_1$). Our matching pipeline proceeds in four steps shown in Figure 4:

1. Compute shape difference operators of dimension $k_\mathcal{M}$ and $k_\mathcal{N}$ for each collection, and aggregate DWKS de-

scriptors for each of them in matrices $A \in \mathbb{R}^{n_{\mathcal{M}} \times np}$ and $B \in \mathbb{R}^{n_{\mathcal{N}} \times np}$.

2. Compute an approximate point-wise map using DWKS descriptors.
3. Project the point-wise map into a low dimension functional map, using only a subset of the vertices.
4. Refine the functional map using *e.g.* the ZoomOut [25] algorithm.

In the first step, we use p evenly spaced energy values $(e_1, \dots, e_p)$ and compute descriptors using Algorithm 2.

In the second step, we firstly combine the standard functional map pipeline described in Section 3.2 with the commutativity terms introduced in [39, 5] :

$$\mathbf{C}^* = \underset{\mathbf{C} \in \mathbb{R}^{k_{\mathcal{N}} \times k_{\mathcal{M}}}}{\operatorname{argmin}} \quad E_d(\mathbf{C}) + \mu_{dc} E_{dc}(\mathbf{C}) + \mu_l E_l(\mathbf{C}) \\ + \mu_c E_c(\mathbf{C}) + \mu_a E_a(\mathbf{C}) \tag{8}$$

with $E_d(\mathbf{C}) = \|\mathbf{CA} - \mathbf{B}\|_F^2$ the descriptor preservation term where $\mathbf{A}$ and $\mathbf{B}$ are matrix A and B projected in the spectral basis, $E_{dc}(\mathbf{C})$ promotes commutativity with operators built from individual descriptors described in Section 3.2, $E_l(\mathbf{C})$ the standard commutativity with the Laplace Beltrami Operator $\|\mathbf{C}\Delta^{\mathcal{M}} - \Delta^{\mathcal{N}}\mathbf{C}\|_F^2$, E_c and E_a respectively enforcing commutativity with the conformal and area-based shape difference operators *ie* $\sum_i \|\mathbf{C}\mathbf{R}_i^{\mathcal{M}} - \mathbf{R}_i^{\mathcal{N}}\mathbf{C}\|_F^2$ and $\sum_i \|\mathbf{C}\mathbf{V}_i^{\mathcal{M}} - \mathbf{V}_i^{\mathcal{N}}\mathbf{C}\|_F^2$ where $\mathbf{R}_i$ is the i-th conformal shape difference operator and $\mathbf{V}_i$ the i-th area one. The result $\mathbf{C}^*$ from problem (8) is then transformed into a point-wise map $T^F : \mathcal{N} \to \mathcal{M}$ using standard method from [28].

In Step 3., we seek to project the point-wise map T into a low-dimensional functional map. To do so we fist discard the fraction α of vertices of $\mathcal{N}$ with the largest descriptor distance defined for vertex j as $d(j) = \|l_{T^F(j)}(A) - l_j(B)\|^2$ where l_m denotes the m-th line of a matrix. This usually ignores vertices near cuts and holes where descriptors are less precise, as seen on Figure 4.

In step 4 we refine the low-dimensional functional map using the ZoomOut algorithm [25]. Note that the absence of refinement algorithm tailored for partial matching makes results particularly sensitive to the refinement parameters. During the first iterations we ignore vertices of $\mathcal{N}$ belonging to the previous subsample and use the complete set of vertices for the last few iterations.

5. Experiments

Parameters. Unless stated otherwise, the parameters for DWKS are fixed across all experiments. The energy values are set to 200 values linearly-spaced values between $-\log 3$ and $\log 3$. The standard deviation parameter σ is set to 1.2% of the total range. The size of the computed shape difference operators is set to $k_{\mathcal{M}} = k_{\mathcal{N}} = 50$ and the functional map used to compute them are of size $3k_{\mathcal{M}} \times k_{\mathcal{M}}$ as advocated in [5]. Parameters for optimization problem (8) are $\mu_{dc} = 10$, $\mu_l = 0$, and $\mu_a = \mu_c = 10^{-4}$. All the terms of Equation (8) have been introduced separately in previous works [28, 27, 5], and we refer the reader to these articles or to the supplementary material for a more in depth discussion on their effect. The low-dimension functional map is a 15×15 matrix for complete shape, and a $15 \times \lambda 15$ matrix for partial shape with λ the estimated slope of the slanted diagonal of the functional map as described

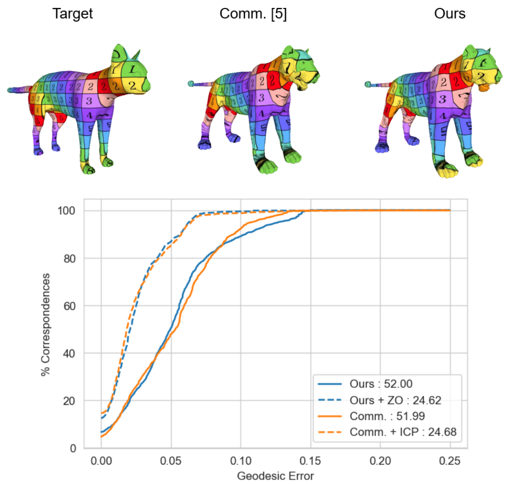

Figure 5. Results on the Sumner dataset using the complete shapes. We show our method obtains similar results as [5] in this case.

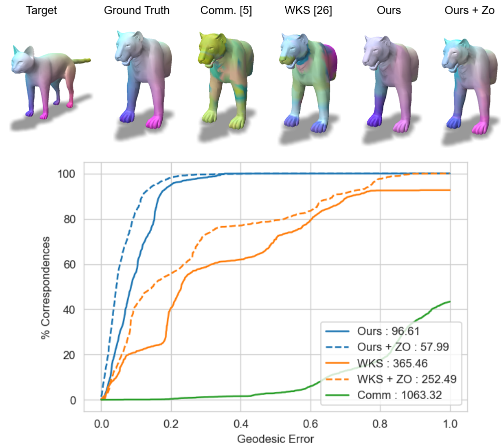

Figure 6. Results on the Sumner dataset. While our method doesn't achieve visually perfect result due to the absence of tailored refinement, it outperforms usual methods. Digits on the legend describe the average geodesic error for each method.

in [33]. We set $\alpha = 20\%$ in the case of partial shapes, and $\alpha = 5\%$ in the case of complete shapes were the amount of noise is reduced. More details about the parameters values can be found in the supplementary material. The implementation our method and the baselines are available at https://github.com/RobinMagnet/DWKS.

Cats and Lions. This first experiment uses synthetic data to evaluate the stability of our method in the standard settings used in [5], and show how our pipeline can handle partiality where the matching technique from [5] might struggle. The two collections consist of 10 similar versions of a cat and a lion meshes as those displayed on Figure 2. We also manually create a collection of lions cut in half, as seen in Figure 6. Using standard parameters and subsampling 1 out of 3 descriptors for faster computation, Figure 5 shows our method achieves similar accuracy than [5] on complete shapes without the need for a costly SVD solver. Figure 6 displays our results in the case of partial before and after the refinement step, compared to those from [5] where we set $\mu_l = 0$ in their objective (5) for fairness since the near-isometry assumption fails. We additionally show results obtained by the standard functional map pipeline [27] using WKS descriptors, described in Section 3.2.

Synthetic face dataset. We use a similar setting on another synthetic dataset [35] consisting of two collections with 10 faces with multiple expressions. As we wish to focus on real noisy scans in the following experiment, we refer the reader to the supplementary material for illustrations of results on this dataset.

DFaust. We finally tested our pipeline on the DFaust dataset [3], which consists of multiple similar collections of real scans of human shapes, which we see as point clouds. This dataset is especially challenging since real data contains notable holes and outlier vertices, which forces us to use approximate intra-collection maps. Using a recent formulation of a Laplacian for point clouds [41], we apply our complete pipeline to collections of complete and partial shapes. Note that the method from [5] can be similarly adapted to work with point clouds and still serves as a baseline. In the case of partial shapes, we again do not apply ICP refinement to results from [5] for fairness. A pointwise map obtained when matching the two collections of humans in jumping motion are shown in Figure 1. In Figure 7 we provide both a qualitative and quantitative evaluation. Our pipeline brings significant improvement to [5] both in the complete and partial setting even without the refinement step, which demonstrates its robustness to noise and applicability to real scenarios. Results from [5], by contrast, do not achieve satisfying results even in the isometric case. Additional quantitative and qualitative results on this dataset as well as comparisons to other baselines are available in the supplementary material.

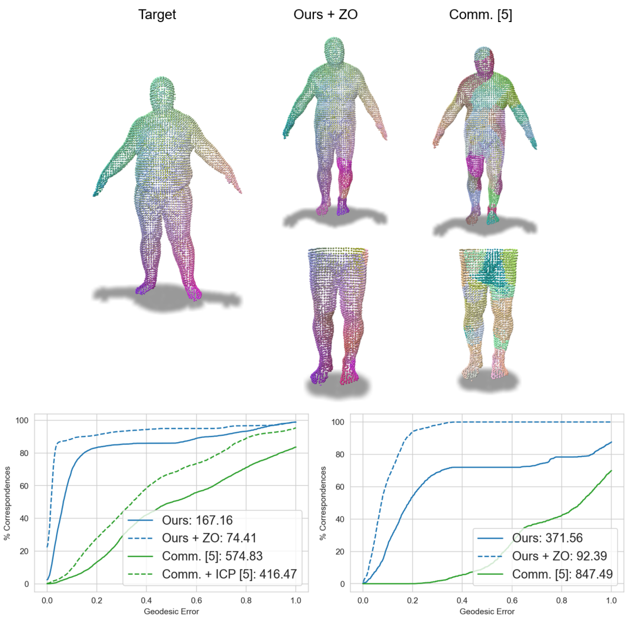

Figure 7. Results on the DFaust dataset. We match two collections of 6 meshes representing humans in jumping motion. Bottom row represents accuracy curves of pointwise maps in the complete (left) and partial (right) cases. Numbers in the legend give the average accuracy multiplied by 10^3.

6. Conclusion and future work

In this work we introduced a pointwise descriptor of deformation between surfaces, able to efficiently encode information about local distortion within a collection at a vertex-level. Our pipeline enables to leverage the common deformations of meshes and point clouds to compute maps in challenging scenarios including symmetry and partiality.

Our approach however suffers from some limitations, as it only focuses on intrinsic deformations of shapes. Furthermore the absence of robust refinement algorithms in the case of partial shapes makes our method very sensitive to the parameters of these algorithms. Finally while the parameters were set as constants across our experiments, they still might have to be manually set by the user.

In the future it will be interesting to exploit meaningful *extrinsic* shape difference operators in the vein of [19], and to potentially overcome the choice of a base shape through the introduction of consistent latent spaces [16].

Acknowledgments The authors thank the anonymous reviewers for their valuable comments and suggestions. Parts of this work were supported by the ERC Starting Grant No. 758800 (EXPROTEA) and the ANR AI Chair AIGRETTE.

References

[1] Noam Aigerman and Yaron Lipman. Hyperbolic orbifold tutte embeddings. *ACM Transactions on Graphics*, 35(6):1–14, Nov. 2016. 1

[2] Mathieu Aubry, Ulrich Schlickewei, and Daniel Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 1626–1633, Barcelona, Spain, Nov. 2011. IEEE. 2, 3, 5, 6

[3] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Dynamic FAUST: Registering Human Bodies in Motion. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5573–5582, Honolulu, HI, July 2017. IEEE. 8

[4] Michael M. Bronstein and Iasonas Kokkinos. Scale-invariant heat kernel signatures for non-rigid shape recognition. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1704–1711, San Francisco, CA, USA, June 2010. IEEE. 5

[5] Aharon Cohen and Mirela Ben-Chen. Robust Shape Collection Matching and Correspondence from Shape Differences. *Computer Graphics Forum*, 39(2):555–568, May 2020. 1, 2, 3, 4, 5, 6, 7, 8

[6] Etienne Corman. Functional Characterization of Intrinsic and Extrinsic Geometry. *ACM Transactions on Graphics*, page 17, 2016. 2

[7] Etienne Corman. *Functional representation of deformable surfaces for geometry processing*. Theses, Université Paris Saclay (COmUE), Nov. 2016. 5

[8] Luca Cosmo, Giorgia Minello, Michael Bronstein, Luca Rossi, and Andrea Torsello. The average mixing kernel signature. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 1–17, Cham, 2020. Springer International Publishing. 2

[9] Nicolas Donati, Abhishek Sharma, and Maks Ovsjanikov. Deep Geometric Functional Maps: Robust Feature Learning for Shape Correspondence. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8589–8598, Seattle, WA, USA, June 2020. IEEE. 1, 2

[10] Danielle Ezuz and Mirela Ben-Chen. Deblurring and Denoising of Maps between Shapes. *Computer Graphics Forum*, 36(5):165–174, Aug. 2017. 2

[11] Danielle Ezuz, Justin Solomon, and Mirela Ben-Chen. Reversible Harmonic Maps between Discrete Surfaces. *ACM Transactions on Graphics*, 38(2):15:1–15:12, Mar. 2019. 1, 2

[12] Maolin Gao, Zorah Lähner, Johan Thunberg, Daniel Cremers, and Florian Bernard. Isometric multi-shape matching. *arXiv preprint arXiv:2012.02689*, 2020. 2

[13] Anne Gehre, Michael Bronstein, Leif Kobbelt, and Justin Solomon. Interactive curve constrained functional maps. In *Computer Graphics Forum*, volume 37, pages 1–12. Wiley Online Library, 2018. 3

[14] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. 3d-coded: 3d correspondences by deep deformation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 230–246, 2018. 1

[15] Qixing Huang, Fan Wang, and Leonidas Guibas. Functional map networks for analyzing and exploring large shape collections. *ACM Transactions on Graphics*, 33(4):1–11, July 2014. 2

[16] Ruqi Huang, Panos Achlioptas, Leonidas Guibas, and Maks Ovsjanikov. Limit Shapes – A Tool for Understanding Shape Differences and Variability in 3D Model Collections. *Computer Graphics Forum*, 38(5):187–202, Aug. 2019. 2, 8

[17] Ruqi Huang, Frédéric Chazal, and Maks Ovsjanikov. On the Stability of Functional Maps and Shape Difference Operators. *Computer Graphics Forum*, pages 1 – 12, 2017. Publisher: Wiley. 5

[18] Ruqi Huang and Maks Ovsjanikov. Adjoint Map Representation for Shape Analysis and Matching. *Computer Graphics Forum*, 36(5):151–163, Aug. 2017. 2

[19] Ruqi Huang, Marie-Julie Rakotosaona, Panos Achlioptas, Leonidas Guibas, and Maks Ovsjanikov. OperatorNet: Recovering 3D Shapes From Difference Operators. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8587–8596, Seoul, Korea (South), Oct. 2019. IEEE. 2, 8

[20] Ruqi Huang, Jing Ren, Peter Wonka, and Maks Ovsjanikov. Consistent ZoomOut: Efficient Spectral Map Synchronization. *Computer Graphics Forum*, 39(5):265–278, Aug. 2020. 2

[21] A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, K. Glashoff, and R. Kimmel. Coupled quasi-harmonic bases. *Computer Graphics Forum*, 32(2pt4):439–448, 2013. 3

[22] B. Levy. Laplace-Beltrami Eigenfunctions Towards an Algorithm That ”Understands” Geometry. In *IEEE International Conference on Shape Modeling and Applications 2006 (SMI'06)*, pages 13–13, Matsushima, Japan, 2006. IEEE. 3

[23] Or Litany, Tal Remez, Emanuele Rodola, Alex Bronstein, and Michael Bronstein. Deep Functional Maps: Structured Prediction for Dense Shape Correspondence. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5660–5668, Venice, Oct. 2017. IEEE. 1, 2

[24] O. Litany, E. Rodolà, A. M. Bronstein, and M. M. Bronstein. Fully Spectral Partial Shape Matching. *Computer Graphics Forum*, 36(2):247–258, May 2017. 1, 2

[25] Simone Melzi, Jing Ren, Emanuele Rodolà, Abhishek Sharma, Peter Wonka, and Maks Ovsjanikov. ZoomOut: spectral upsampling for efficient shape correspondence. *ACM Transactions on Graphics*, 38(6):1–14, Nov. 2019. 2, 6, 7

[26] Andy Nguyen, Mirela Ben-Chen, Katarzyna Welnicka, Yinyu Ye, and Leonidas Guibas. An Optimization Approach to Improving Collections of Shape Maps. *Computer Graphics Forum*, 30(5):1481–1491, Aug. 2011. 2

[27] Dorian Nogneng and Maks Ovsjanikov. Informative Descriptor Preservation via Commutativity for Shape Matching. *Computer Graphics Forum*, 36(2):259–267, May 2017. 2, 3, 7, 8

[28] Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. Functional maps: a flexible

representation of maps between shapes. *ACM Transactions on Graphics*, 31(4):1–11, Aug. 2012. 2, 3, 7

[29] Maks Ovsjanikov, Etienne Corman, Michael Bronstein, Emanuele Rodolà, Mirela Ben-Chen, Leonidas Guibas, Frederic Chazal, and Alex Bronstein. Computing and processing correspondences with functional maps. In *ACM SIGGRAPH 2017 Courses*, SIGGRAPH '17, New York, NY, USA, 2017. Association for Computing Machinery. 3

[30] Jing Ren, Simone Melzi, Maks Ovsjanikov, and Peter Wonka. MapTree: recovering multiple solutions in the space of maps. *ACM Transactions on Graphics*, 39(6):264:1–264:17, Nov. 2020. 1, 2

[31] Jing Ren, Adrien Poulenard, Peter Wonka, and Maks Ovsjanikov. Continuous and orientation-preserving correspondences via functional maps. *ACM Transactions on Graphics*, 37(6):1–16, Jan. 2019. 1, 2, 3

[32] Martin Reuter, Franz-Erich Wolter, Martha Shenton, and Marc Niethammer. Laplace–Beltrami eigenvalues and topological features of eigenfunctions for statistical shape analysis. *Computer-Aided Design*, 41(10):739–755, Oct. 2009. 3

[33] E. Rodolà, L. Cosmo, M. M. Bronstein, A. Torsello, and D. Cremers. Partial Functional Correspondence: Partial Functional Correspondence. *Computer Graphics Forum*, 36(1):222–236, Jan. 2017. 1, 2, 3, 6, 8

[34] Jean-Michel Roufosse, Abhishek Sharma, and Maks Ovsjanikov. Unsupervised Deep Learning for Structured Shape Matching. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1617–1627, Seoul, Korea (South), Oct. 2019. IEEE. 1, 2

[35] Raif M. Rustamov, Maks Ovsjanikov, Omri Azencot, Mirela Ben-Chen, Frédéric Chazal, and Leonidas Guibas. Map-based exploration of intrinsic shape differences and variability. *ACM Transactions on Graphics*, 32(4):1, July 2013. 2, 3, 5, 8

[36] Yusuf Sahillioğlu. Recent advances in shape correspondence. *The Visual Computer*, 36(8):1705–1721, 2020. 1

[37] Yusuf Sahillioğlu. Recent advances in shape correspondence. *The Visual Computer*, 36(8):1705–1721, Aug. 2020. 2

[38] Patrick Schmidt, Janis Born, Marcel Campen, and Leif Kobbelt. Distortion-minimizing injective maps between surfaces. *ACM Transactions on Graphics*, 38(6):1–15, 2019. 1

[39] Nitzan Shapira and Mirela Ben-Chen. Cross-Collection Map Inference by Intrinsic Alignment of Shape Spaces. *Computer Graphics Forum*, 33(5):281–290, Aug. 2014. 1, 2, 3, 4, 5, 7

[40] Abhishek Sharma and Maks Ovsjanikov. Weakly Supervised Deep Functional Maps for Shape Matching. *Advances in Neural Information Processing Systems*, 33:19264–19275, 2020. 2

[41] Nicholas Sharp and Keenan Crane. A Laplacian for Nonmanifold Triangle Meshes. *Computer Graphics Forum*, 39(5):69–80, Aug. 2020. 2, 8

[42] Yanyao Shen, Qixing Huang, Nati Srebro, and Sujay Sanghavi. Normalized spectral map synchronization. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 2

[43] Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion. *Computer Graphics Forum*, 28(5):1383–1392, July 2009. 2, 3, 5

[44] Oliver Van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. A survey on shape correspondence. *Computer Graphics Forum*, 30(6):1681–1707, 2011. 1

[45] L. Wang, A. Gehre, M. M. Bronstein, and J. Solomon. Kernel Functional Maps. *Computer Graphics Forum*, 37(5):27–36, Aug. 2018. 2, 3